

研究ノート

心理データ処理演習における Word 課題の自動採点処理プログラムの作成と適用

中京大学心理学部非常勤講師 重村 正之

Create and apply automatic scoring program for the Word task by using the VBA in Date-processing exercises of psychology

SHIGEMURA, Masayuki (Chukyo University)

Office tasks were analyzed for automatic scoring process using Word VBA. The file created by the Word analyzed in VBA code. Analysis contents are as follows. 1. Or page setup has been run correctly. 2. Use the field code, to confirm whether the setting of the ruby has been performed correctly. Here, the ruby refers to the character that is displayed on top of the Chinese characters. Also showed display code of how the dialog box.

Key words: Automatic scoring program , Word VBA , field code ,Dialog box , WdStatistic

1 はじめに

心理学部の学生は、Microsoft Office を用いてレポートや卒業論文を作成している。このレポートや卒業論文は、科学論文の体裁を整えている必要がある（例えば、科学論文での本文、表、グラフの作成方法には学会の規則や手引きが存在する）。この手引き等に従えるだけの Word や Excel の知識や技能を習得していくことが必要となる。特に、心理学で必要な手続きや手法は、他の分野と異なる。例えば、データ処理の際に、標準偏差 (SD) を求めるが、使用する関数が理工系とは異なる。

ところで、一般にある教科（例えば心理学概論等の講義科目）が十分に理解できたか否かを検証するには、学習の到達目標に対する評価をする為に、通常、検証問題を学生に課し、結果の採点をし、効果判定をおこなう。この場合検証課題（レポート等）の判定では、課題内容の検証（レポートのチェック）を評価者が熟読すれば行える。しかし、Word や Excel の知識や技能が習得されているか否かを判定する為には、検証課題の結果と操作との比較が必要となる。この比較（採点作業）は、作業手順が複雑で時間や手間がかかり、指示通りに行えたか否かの採点基準の一部を見落したり間違えたりすることも十分考えられる。この採点作業が、ある程度自動化できれば、より高度の効果判定ができると考えられ

る。そこで、今回は、自動化がどの程度行え、効果の判定がどのように行えるかを検証する。

ここでは、Word 文章の作成手続や構成方法等に対する教育効果の判定のために、対応する事項の確認用の課題を作成し、その結果を解析することで判定する。

本論文では、Word の VBA (Visual Basic for Application (以下 VBA と略す)) のプログラムを作成し、採点の自動化が何処まで可能なかと教育効果の確認をする。さらに、教育効果の判定用に作成された課題の解析プログラムの問題点のいくつかを取り上げる。ここで見いだされた問題点を改良し、より良い VBA プログラムを作成することにより、授業の改善に用いることができるようになると思われる。

2 課題解析のための方法について

課題解析のために VBA を使用する。VBA は、Microsoft Office の各製品の使いやすさを向上するために開発されたプログラム言語である。VBA を利用することにより、Excel や Access 等の定型業務を自動化することができる（月ごとや週ごとの定型的な処理を毎回最初から作業しなくてもよい）。今回は、Word 課題の解析の為に Word の VBA のプログラムを作成した。作成された課題を解析する

ことにより、指示通りに課題が処理されているのかがどうか明らかにできる。指示通り処理が行えていなかったということは、時間が不足していたか理解や操作方法が十分に習得されていなかったことが考えられる。これらを調べることにより、改善すべき点を見いだせる。さらに、数多くの課題解析の際、見落としや基準の変化などをなくすことにも、VBAによるプログラムは役に立つ。なお、VBAは、Microsoft Officeのバージョンが2013より前と後では、命令語に変更があることが判明した。

3 プログラムの概要

本論で取り上げるプログラムを作成するコンピュータの環境として、OS: Windows 10, アプリケーションは、Microsoft Office 2013を使用した。

Wordの課題の概略として、1) 入力後、ファイルの挿入を行い、ページ設定として、文字数と行数、余白の変更を行うこと。2) 特定の文字列に、ルビをふること。3) 特定の文字列に対し、フォント名の変更とフォントサイズの変更を行うこと。4) 表を作成し、特定のセルの文字列に対し、隠し文字の設定を行うこと。5) セクションを1つ増やし、新しい設定として、用紙サイズの変更、文字数、行数、上下左右の余白の設定と印刷の向きを行う、の5個を設定し、これらを確認するWord VBAを記述した。

4 Wordの具体的課題

今回設定したWordの課題は、心理学についての文章に対する設定を問った問題を作成した。Figure 1に見本を示す。

Word課題では以下の指示を与えた。

- (1) 「1番目の解答」から「問題文」までの文章を入力し、次に、「問題文」の次の行に、「問題文 - 性格特性論.docx」を挿入し、その後、学籍番号 - 特性論で保存しなさい。
- (2) 余白は上下左右を25mmとし、ページの文字数を43に、行数を43に変更しなさい。
- (3) 文章の下に、1行あけて、「解答群」と入力し、その下に、解答群を2行で作成し(本文よりコピー)、語と語の間は「カンマ(,)」で区切り、文字の先頭とカンマの位置は、ほぼ揃っていること。解答群の「性格特性」に対し、カタカナ

でサイズ6ptのルビを振りなさい。

- (4) 解答群の下に1行あけて「解答欄」と入力し、解答欄(4行5列)を作成しなさい。
- (5) 解答欄の1行目のセルには から と入力し、高さを6mmに指定しなさい。次に、2行目は から の解答を本文よりコピーし、高さを15mmに設定しなさい。3行目は から と入力し、高さを6mmに設定し、4行目は から の解答を本文よりコピーし、高さを15mmに設定しなさい。なお、解答欄の文字位置は、すべて「中央揃え」とする。高さは、すべて「固定値」に設定する。ルビはなし。
- (6) 本文中の括弧の中の答と解答欄の答に、隠し文字の設定をしなさい。
- (7) 解答欄の下に、「次のページから開始」のセクション区切りを挿入し、ファイルを上書き保存しなさい。
- (8) 2ページ目を次のページ設定に変更しなさい。用紙サイズをB5とし、印刷の向きは横とし、上下左右の余白をすべて20mmとし、文字数を60文字とし、行数を28行に設定しなさい。

5 教育効果判定用の確認事項

- (1) ファイルが正しいファイル名で保存されていること。
- (2) ページ設定が正しく行われていること。
- (3) 1行開けて、「解答群」が入力されていること。コピーが正しく行われていて、語と語の間が「カンマ(,)」で区切られていること。特に、カンマの区切りが「点」ではなくカンマに変更ができていないこと。文字の位置が、正しい設定になっていること。ルビが正しく設定されていること。
- (4) 1行開けて「解答欄」と入力されていること。
- (5) 4行5列の表が作成されており、セルに正しく語がコピーされていること。セルの高さが正しく設定されており、文字位置が「中央揃え」に設定されていること。
- (6) 隠し文字設定が本文と解答欄に正しく行われていること。
- (7) セクション区切りが正しい位置に設定してあること。
- (8) 2ページ目のページ設定が正しく行われていること。

1 番の解答

問題1 次の文章の空欄にもっとも適切な語句を下記の解答群より選び、解答欄に記入しなさい。

問題文

性格特性論

性格特性論に基づいて作成された矢田部・ギルフォード(Y-G)性格検査全には、120項目ある。各項目は、(① 抑うつ性)、気分の変化、劣等感、神経質、主観性、非協調性、(②), 活動性、のんきさ、思考的外向、支配性、社会的外向という12の(③)に対応している。

個々の性格の要素を判断する

特性論とは、性格を(④)多くの特性(その人を特徴づける一定の行動パターン)に分け、個々の(⑤)についてそれぞれに判断し、それらを総合して全体として性格を描写しようとする立場である。複雑な人間の性格を2~3のタイプに大雑把に分けざるをえなかった(⑥)にくらべると、性格の個々の要素について客観的に判断できるため、より緻密な性格診断が可能となるのである。

特性は類型を構成する要素である

アイゼンクは、特性論に因子分析法を導入して、内向性-外向性次元、神経症的次元、精神病的次元という(⑦)の相互に独立した次元を抽出し、これらを類型(タイプ)として考えた。性格特性は類型を構成するもので、たとえば外向性だと、活動性や社会性、冒険心、衝動性、表現力、無反省、無責任の7特性からなっている。これらの特性は、さらに習慣的反応の水準、特殊的反応の水準の2水準に分れ、類型を(⑧)とした階層構造をなしているものと性格をとらえたのである。

しかし、彼の類型は、外向でなければ内向であるといった古典的な(⑨)的発想には立っていない。科学的な特性論の立場から類型論の良さも取り入れた総合的な立場にある理論といってもよいであろう。この考えに基づいて(⑩)性格検査が生れている。Y-G検査も、このような流れの中から作られた性格検査である。

解答群

抑うつ性 , 攻撃性 , ^{セイカクトクセイ}性格特性 , 比較的 , 特性
 類型論 , 3つ , 頂点 , 二分法 , アイゼンク

解答欄

④	②	③	④	⑤
抑うつ性				
⑨	⑩	⑧	⑨	⑪

Figure 1 Word の課題出来上がり見本の一例

・なぜ隠し文字の設定が必要か

フォントの「隠し文字の設定」が行われていると、ホーム タブの段落グループの上の段の右端にある編集記号の表示/非表示を非表示に設定することにより、Figure 1の()から()のように、解答が表示されないが、表示にすると、(抑うつ性)のように表示される。ここでは、違いを示すために、()とセルの下には、隠し文字の設定が解除してある。隠し文字の設定は、1つのファイル(文章)で、問題用紙と解答用紙とを同時に作成することができる便利な設定である。Figure 1に示されるように、隠し文字の設定がされていると、印刷時にその部分は印刷されないで、そのまま問題用紙になる。解答も含まれているので、隠し文字を印刷する設定(ファイル オプション 表示の隠し文字を印刷するという設定)で印刷すると、解答用紙として、印刷される。印刷が終わったなら、隠し文字の印刷をする設定を解除しておく。

6 Word VBA プログラムについて

Word VBA を用いた解析作業は、教育効果判定用の確認項目が正しく設定されているか否かを調べるものである(例えば、用紙のサイズがB5に設定されているかの確認を、レイアウト ページ設定をクリックして調べなくても確認できる)。しかし、Word VBA は、資料が少ないので、プログラム作成には幾つかの補助手段が必要となった。その1つとして、自動記録マクロがある。マクロを使用する為には、リボン(Microsoft Office を起動した際の画面上部に「ファイル、ホーム、挿入等」が表示される部分があるが、その部分を Microsoft はリボ

ンと呼んでいる)に「開発」タブ(購入時に開発という表示が現れていない)を表示させる。

この手順として、ファイル オプション リボンのユーザー設定 右側の開発にチェックを入れる。次に、セキュリティに関する変更を加える。「ファイル オプション セキュリティセンター セキュリティセンターの設定 マクロの設定 すべてのマクロを有効にする」の順に設定し、VBA プロジェクトオブジェクトモデルへのアクセスを信頼するにチェックの後にOK ボタンを押す。この変更を加えないと、VBA が動作しない。

次に Visual Basic Editor で、プログラムを作成するために、コードウィンドウを開く。操作として、挿入 標準モジュールをクリックする。

Figure 2 にコードウィンドウの例を示す。マクロの自動記録を実行しても、作成される。

7 プログラムの基本形

VBA のプログラムは最初の行と最後の行の形が決まっている。

```
Sub プログラム名 ( )
例   Sub Macro 1 ( )
必ずプログラム名の後の左右の括弧が必要
    .
    .
プログラム本体
    .
    .
End Sub
```

先頭の Sub に続く「プログラム名」は日本語が使用できる。自動記録マクロで記録前につけたマクロ名は、この部分に自動的に書き込まれる。プログラム名の後の左右の括弧の間には、通常何も書かないことが多い。Visual Basic Editor では、先頭行「Sub プログラム名 ()」の最後のカッコを書いた後、Enter キーを押すと、自動的に2行下に、「End Sub」という行が挿入される。VBA の命令語は、基本的に半角文字を使用のこと。Visual Basic Editor に複数のプログラム(Sub.....End Sub)を書き込むと、プログラムとプログラムを区切る横線が自動的に挿入される。なお、作成したプログラムを直接保存する必要はない。プログラムは、原則 Word の文章に保存されるので、文書を保存

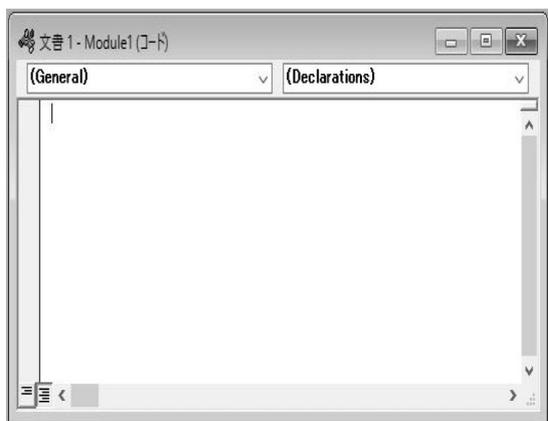


Figure 2 ワードのコードウィンドウの例

するときに、「マクロ有効文書 (拡張子が *.docm となる)」で保存する。通常の形式 (*.docx) で保存すると、マクロや VBA は保存されないの注意が必要である。

7-1 Visual Basic Editor に付加するウィンドウ

Visual Basic Editor だけでは、使用しにくいので他のウィンドウを付加する。付加するウィンドウには、イミディエイトウィンドウ、ローカルウィンドウ、ウォッチウィンドウとする。さらに、オブジェクトブラウザー (F2 キーを押すことでも表示できる) を用いると、VBA の命令 (プロジェクトで使用できるシンボル (命令語)) を確認できる。これらは、表示をクリックして表示させる。さらに、表示 ツールバーにある標準と編集にチェックを入れておく。

イミディエイトウィンドウは、式のデバックと評価、ステートメントの実行、変数値の出力などのために使用する。ローカルウィンドウでは、ブレークポイントまたは「STOP」で停止させた場合のプログラム中に使用されている変数の現在の値を確認することができる。[ウォッチ] ウィンドウを使用すると、変数と式を評価できる。

オブジェクトブラウザーを使用すると、プロジェクトで使用できるシンボル (命令語) を調べることができる。

8 プログラム概略

8 つの教育効果判定用の確認項目について、プログラムを作成した。

確認項目(1)のファイルが保存されているか、につ



Figure 3 マクロの記録ダイアログボックス

いては指定の文字入力がないと、Word ファイルが挿入された文章が指定ファイル名で保存されているのかと、ファイル中の文字数が一定の値以上になっているかを調べることで確認する。

プログラムを作成するために、マクロで手続きを実行し、記録する。

・マクロの記録方法

最下段にあるステータスバーの中央付近にある「新しいマクロの記録を開始するボタンをクリックする」。これは、タブ「開発」のコード領域にある「マクロの記録」をクリックしても同じことができる。クリックしたことによりマクロの記録ダイアログボックスが出現する (Figure 3 参照)。

Figure 3 に示すように、マクロ記録のダイアログボックスが表示されるので、上段にあるテキストボックスにマクロ名 (プログラム名) を指定する。指定しなくても可能である。

次に、マクロの保存先 (S) について、これは必ず指定する。現在、すべての文書 (Normal.dotm) になっているが、このままでは、すべての文書に保存されるので、「現在の文書」に必ず変更する。マクロを割り当てる対象 (ボタンとキーボード) は指定しない。説明には、操作を記述しておく、それがプログラム中に注釈行として保存されるので、書いておくと良いがなくても可能である。ファイルをオープンするマクロを記録し、Figure 3 を表示させ、マクロの保存先を変更して、「OK」ボタンをクリックする。次に、タブのファイル 開くから、必要なファイルを開く。必要なファイルがオープンされたなら、マクロを停止させる。マクロの開始ボタンが、終了ボタンに変化しているので、これをクリックする。

作成されたマクロを確認するには、Visual Basic Editor 画面の左にあるプロジェクト - Project にある Project (ファイル名) の下の標準モジュールをダブルクリックし、NewMacroes を選択する。右側に、コードウィンドウが表示される。

・コードの表記について

コードは、基本的に、予約語とドット (.) 予約語のパターンで記述する。予約語の後に左括弧があり、引数を必要とする場合は、引数と引数の間は、カンマ (,) で区切る。

8-1 ファイルのオープン

自動記録マクロで作成したコードの結果例

```
ChangeFileOpenDirectory "C:\¥Users¥
ここにはファイルのパスが記述されている"
Documents. Open FileName:="ファイル名.
docx" , ConfirmConversions:= False _
, ReadOnly := False ,
AddToRecentFiles :=False ,
PasswordDocument :="" , _
PasswordTemplate :="" , Revert :=False ,
WritePasswordDocument :="" , _
WritePasswordTemplate :="" ,
Format :=wdOpenFormatAuto ,
XMLTransform :=""
```

このマクロは、ファイルの存在している場所を指定し（ファイルのパスを指定し）、ファイル名を設定しないと、動作しない。ファイルのパスやファイル名をプログラムの中に記述しないでファイルを開くために、ダイアログボックスを使用するプログラムに変更する。このプログラムは、複数のファイル選択が可能な場合である。

```
Dim dlgOpen As FileDialog
・FileDialog 型の変数を定義している
Set dlgOpen = Application . FileDialog( _
・行の最後（スペースと__）は、継続行の指定である
FileDialogType:=msoFileDialogOpen)
・定義された変数（dlgOpen）に値をセット
With dlgOpen
. AllowMultiSelect = True
・複数ファイルの設定を可とする
先頭（Allo・・・の左）にドットを付加する
. Show
・ダイアログボックスを表示する
先頭（showの左）にドットを付加する
End With
同等な命令として
Dialogs( wdDialogFileOpen ) . show・・・
( ) で置き換えられる。
```

マクロから得られたプログラムには、オプション項目が相当数あるが、すべて必要ではない。また、今回はマクロの結果ではなく、() を使用した。

・プログラム記入上の注意
予約語（命令語のこと）は、小文字で入力しても、正しく入力されていると、一部が、大文字に変更される。もし、Enter キーを押しても、大文字にならない場合、スペルミスの可能性が高い。その際、入

力途中で CTRL + スペースを押すと、正しい綴りが表示されることがある。また、予約後の最後に、ピリオド（.）を入力すると、次に使用可能な予約後の一覧が表示されることもある。

プログラムが、横に長い場合に、行の最後に半角スペース+_（スペースとアンダーバー）を入力し、エンターキーを押すと、次の行に継続して、プログラムが記述できる（マクロの例を参照）。

8-2 文章を行（センテンス）単位で配列変数に取り込む方法

プログラム

```
Sub センテンスごとに配列に格納する( )
Dim str() As String
・配列変数 str を予約する。大きさを ReDim で決定する
Dim cnt As Long
・変数 cnt を予約した
Dim i As Long
・変数 i を予約した
ActiveDocument . ActiveWindow . View .
ShowFieldCodes = Not _ ActiveDocument .
ActiveWindow . View . ShowFieldCodes
・フィールドコードを表示
With ActiveDocument . Sentences
cnt = . Count
・オープンされている文章の行数をカウントした
ReDim str(1 To cnt)
・配列の大きさを1からの形で定義した
For i = 1 To cnt
str(i) = . Item(i) . Text
・行を文字列として取り込んだ
Next i
End With
ActiveDocument . ActiveWindow . View .
ShowFieldCodes = Not _ ActiveDocument .
ActiveWindow . View . ShowFieldCodes
・フィールドコードを非表示にした
End Sub
フィールドコードは、ルビを確認するために必要な
ので、ここでは、挿入しているが、なくてもよい。
```

8-3 文字数のカウント

ファイルが読み込まれたら、文字数をカウントして、入力と文章の挿入が行われたかを確認する。入

力文字数は、52 文字なので、文字数が 52 以上であり、4 つめの文章 (センテンス) が「性格特性論」なので、4 番目のセンテンスと比較して確認し、同じなら挿入がなされたと判定する。

文字数をカウントするコードの構文は、
ActiveDocument .
ComputeStatistics(Statistic:=wdStatisticWords ,

—
IncludeFootnotesAndEndnotes :=True)

構文は、
* . ComputeStatistics (Statistic ,
IncludeFootnotesAndEndnotes)

Statistic は必須で、WdStatistic 型
IncludeFootnotesAndEndnotes はオプションでバリエーション型 (True なら脚注と文末脚注を含める、規定値は false で省略可) なので、
n=ActiveDocument .
ComputeStatistics(Statistic:=wdStatisticWords)
とする。

wdStatisticWords の部分は、Table 1 に示す別の予約語に変更できる。変更値は、名前の代わりに値を指定できる。これらはイミディエイトウィンドウで確認できる。

8-4 ページ設定

次に、上下左右の余白を 25mm とし、ページの文字数を 43 に、行数を 43 が、正しく設定されているかの確認方法について述べる。

ここでは、カーソルを先頭ページに移動し、ページ設定について確認した。ページ設定では、9 種類の項目の確認をする。そこで、それぞれの値を格納するために、9 個の要素を持つ配列を定義した。今回は、2 回ページ (2 種類の) 設定確認するので、配列は、2 次元とする。変数の型は Variant とする。

コード
Dim pag_set(1 , 9) as Variant
Selection . GoTo what:=wdGoToPage ,
which:=wdGoToFirst
・先頭ページに移動
With Selection . PageSetup
pag_set(0 , 1) = . CharsLine
・文字数がセットされた
pag_set(0 , 2) = . LinesPage
・行数がセットされた
pag_set(0 , 3) = . TopMargin
・上余白がセットされた
pag_set(0 , 4) = . BottomMargin
・下余白がセットされた
pag_set(0 , 5) = . LeftMargin
・左余白がセットされた
pag_set(0 , 6) = . RightMargin
・右余白がセットされた
pag_set(0 , 7) = . LayoutMode
・レイアウトモード：0 なら標準、1 なら文字数と行数指定、2 なら行数だけ指定、3 なら原稿用紙の指定がされている。
pag_set(0 , 8) = . PaperSize
・ページサイズは A3 なら 6 が、A4 なら 7 が、B4 なら 10 が、B5 なら 11 が設定されている。
pag_set(0 , 9) = . Orientation
・印刷の向きは、0 なら縦が、1 なら横が設定されている。
End With
注意) すべての「=」の後にドット (.) がある。
例えば、文字数の場合は、
= . CharsLine である。これは、Selection . PageSetup . CharsLine を With と End With を用いて記述してある。
語がコピーされているか、カンマの位置合わせに

Table 1 変更値 (WdStatistic 列挙 (Word))

名 前	値	説 明
wdStatisticCharacters	3	文字数
wdStatisticCharactersWithSpaces	5	スペースを含めた文字数
wdStatisticFarEastCharacters	6	アジア言語の文字数
wdStatisticLines	1	行数
wdStatisticPages	2	ページ数
wdStatisticParagraphs	4	段落数
wdStatisticWords	0	単語数

については、目視で行う。

8-5 ルビの確認

ルビを確認する為に、フィールドコードを表示し、確認する。

ここでは、「性格特性」に対し、カタカナでサイズ6ptのルビを振る。
セイカクトクセイ
 性格特性 となる。

・ルビに使用されているフィールドコードの説明
 すべてのフィールドコードを表示/非表示を切り替えるために ALT + F 9を押す。

性格特性の場合は、
 { EQ ¥ * jc2 ¥ * "Font:M S 明朝" ¥ * hps 10 ¥ o ¥ ad(¥ s ¥ up 9(セイカクトクセイ) , 性格特性) }
 と表示される。

・EQ フィールドは数式を作成する場合に使用できる。また、文字飾りとして利用もできる。通常、ルビは、ルビのダイアログボックスで設定する。

設定を変更した際のフィールドコードを比較すると、配置を変更すると jc0~jc4 の部分(上述では jc2) が変わり、¥o の後に ¥ad (均等)、¥al (左揃え)、¥ar (右揃え) が付加される。フォントを変更すると Font にフォント名が変更され、() 内のルビのフォントが変更される。オフセットを 0 pt から 5 pt に変更すると ¥up 9 の部分が ¥up 14 に変わり、サイズを 20pt に変更すると ¥ * hps 10 の部分が ¥ * hps 20 に変わる。

8-5-1 スイッチについて

¥ * 書式スイッチを意味する。

jc 配置を指定する。

指定の値は、

jc0 中央揃え、jc1 均等割り付け1、jc2 均等割り付け2、jc3 左揃え、jc4 右揃えとなる。

hps はサイズでポイント数に關係して、2倍された値で、10だと、5ptを表す。

¥o 重ね書きを表す。

¥s ¥up シフトする(漢字の上に重ね書きする)

¥s ¥do シフトする(漢字の下に重ね書きする)

ルビ確認部分のみのコード

```
ActiveDocument . ActiveWindow . View .
ShowFieldCodes = Not _ ActiveDocument .
ActiveWindow . View . ShowFieldCodes
```

・フィールドコード表示

```
With ActiveDocument . Sentences
cnt = . Count
For i = 1 To cnt
x = InStr(1 , . Item(i) . Text , "hps" , 1)
If x <> 0 Then
sen = , . Item(i) . Text
ii = i - 1
End If
Next i
End With
x = InStr(1 , sen , "hps" , 1) + 3
hps の h があるのは、x 文字目なので、値の位置は +3 する
Z = Mid(sen , x , 2) / 2
Z は文字サイズで、hps の値は、2倍されている
x = InStr(1 , sen , "up" , 1) + 3
y = Mid(sen , x , 1)
If y = 9 Then
y = 0
Else
y = Mid(sen , x , 2)
・オフセットの値 y は 0 ポイントの時 9 なので -9 する。
y = y - 9
End If
s2 = Mid(sen , x + 2 , 8)
・ルビ文字取得する。
フィールドコードを非表示に戻す。
```

8-6 表の作成

4行5列の表作成とセルの高さと文字位置の設定について

Figure 1 に示すように、解答欄は表形式で作成することが求められている。

表については、該当するドキュメントに幾つ表があるかを調べ、問題とどの表が一致しているのかを確かめる必要がある。通常、表が複数作成されている場合、表の削除や移動をしなければ、作成した順番に表の番号が与えられる。今回は、1つだけ表が作成されている前提で確認する。表が作成されていないのに、次のコードを実行すると、実行時エラーが発生する。

・表の数(ドキュメントの中に表はいくつ作成され

ているか) を数えるコードは
 hyo = ActiveDocument . Tables . Count
 で求められ、このドキュメント中の表数が hyo に
 格納される。

設問の行数と列数で表が作成されているかを調べる
 コードは

```
With ActiveDocument . Tables (1)
Retusu = . Columns . Count
・ Retusu に列数をカウントする。
Gyosu = . Rows . Count
・ Gyosu に行数をカウントする。
```

End With
 今回は、表が 1 つのみなので、Tables (1) と記
 述する。

変数、Retusu と Gyosu が 4 と 5 になっていれ
 ば設定と同じである。

・セルの高さを確認するコードは
 行の先頭セルの高さを行数分確認する。

```
For i = 1 To Gyosu
x = Selection . Rows . Height
・ x に高さを求めている。
```

```
xm = Round(PointsToMillimeters(x), 1)
・高さをのポイントをmm単位に変換している
```

```
xr = Selection . Rows . HeightRule
・サイズの高さの設定値が得られる。
```

xr が 0 なら最小値で、1 なら固定値となる。高さ
 の判定を行うために値を取得する。

```
Selection . Move . Unit:= wdRow , Count:= 1
・1行下のセルに移動
```

Next i
 ・文字の位置が中央揃えかどうかを判定するコード
 表を作成し、文字列の配置を指定する際に、表ツ

ル レイアウト 配置グループの中央揃えボタンを
 クリックすると、同等のコードは以下の通りである。

```
With ActiveDocument . Tables(1)
Selection . ParagraphFormat . Alignment
= wdAlignParagraphCenter (1)
```

```
Selection . Cells . VerticalAlignment =
wdCellAlignVerticalCenter (1)
```

「wdAlignParagraphCenter」は段落の配置が中央
 であること、セルならば中央揃えを示している。
 「wdCellAlignVerticalCenter」はセル内にある文
 字列の垂直方向の配置が中央であることを示してい
 る。

従って、作成された表の文字列の配置を確認するた
 めには

```
tate = ActiveDocument . Tables(1) .
ParagraphFormat . Alignment
yoko = ActiveDocument . Tables(1) .
Cells . VerticalAlignment
```

として、tate が 1 で、yoko が 1 ならば、中央揃え
 の設定がなされている。

8-7 隠し文字設定が行われているか

今回の文章には、本文中の括弧の部分と表の部分
 には、フォントの文字飾りの設定として、「隠し文
 字」を指定する。これは、フォントのダイアログボ
 ックスを使用して、設定ができるかどうかを問う問
 題である。隠し文字の設定が指示されている特定の語
 を選択し、次のコードで確認する。

```
If Selection . Font . Hidden then
```

この部分に対応する処理を記述

```
end if
```

という形である。

Table 2 Range. GoTo メソッドのパラメータと説明

名前	必須か否か	データ型	説明
What	オプション	バリエーション型 (Variant)	範囲の移動先となる項目の種類。WdGoToItem 定数のいずれか (wdGoToTable , wdGoToSection , wdGoToLine) を設定する。
Which	オプション	バリエーション型 (Variant)	範囲の移動先となる項目。WdGoToDirection 定数のいずれかを設定する。
Count	オプション	バリエーション型 (Variant)	文書内の項目の数。既定値は、1 です。正の値のみが有効です。範囲の前にある項目を指定するには、Which として wdGoToPrevious を使用して、引数 Count の値を指定する。
name	オプション	バリエーション型 (Variant)	What の指定が、wdGoToBookmark, wdGoToComment, wdGoToField, または wdGoToObject の場合は、この引数で名前を指定する。

8-8 セクション区切りが設定してあるか

セクションが複数個存在すればいいので、セクションの数を数える。

```
sec_su = ActiveDocument . Section . Count
```

の

sec_su にセクションの数を求め、1以上かの判定で確認する。

8-9 ページ設定 (2 ページ目) が正しく行われているか

今回は、1 ページ目と2 ページ目のレイアウトの設定が異なっている。そこで、2 ページ目のレイアウトを確認する。

2 ページ目に移動するには、
Selection.GoTo What:=wdGoToPage,
Which:=wdGoToAbsolute, Count:=2

What, Which 等はパラメータといい、以下の意味がある (Table 2 参照)。

今回は、2 ページ目の指定なので、
What:=wdGoToPage としたが、wdGoToSection
でも可能である。その後、8-3 に示しているコード
を実行する。

ファイルの解析の後、ファイルを保存する。ファイルを名前を付けて保存するには、
Dialogs(wdDialogFileSaveAs). Show
を用いる。

9 教育効果の判定

今回の Word の問題 (Figure 1 参照) では、約 97% の正答率をほぼ全員が得ていた。このことから、今回の教授法で成果が十分得られることが確認できた。

10 まとめ

Word で作成された課題ファイルの内容を採点する方法の一つとして、Word VBA を用いて、課題の解析を行うプログラムを作成し、その概要を説明した。VBA を使用することにより、教育効果の有無の解析が行えた。Word VBA には、資料がほとんどなかったため、筆者はワードの課題指示の操作を自動記録マクロでコードにし、それを解析し、Microsoft のサイトで調査し、修正する方法でプロ

グラムを作成した。命令語を調査するには、命令語をクリックし、F1 キーをクリックすれば行える。

実際に、Word VBA を適用した結果は、指定した指示のほぼ 97% が守られていた。プログラム上で、指示通りに行われていないと判断されたケースでは、例えば、余白の設定が 1 mm ずれていたり、文字数の設定を飛ばしていたりする場合が見られた。余白の設定が 1 mm ずれていた間違いは、理解不足というより、マウス操作等の操作ミスで生じた可能性があり、理解不足とは考えにくいものである。また、文字数の設定に関しては、2 箇所中の 1 箇所でのみ起きていたので、これも理解不足とは考え難い。

なお、今回示した VBA の範囲では、解析できない分野も存在している。例えば、文字等の塗りつぶしの色の設定の解析では、標準の色とテーマの色では、値の調査がしにくいものであった。また、位置関係の解析の仕方やフィールドコードの解析について、スタイルや箇条書きに対する設定・変更などもさらなる調査が必要であると考えられる。

また、教育効果判定を自動化することは困難な場合が多いが、分野を限定したり効果測定の方法を工夫したりすることによりかなり、自動化ができると考えられる。例えば、今回開発した自動採点プログラムの応用として、Figure 1 に示したような解答欄に答を記入する方式で結果が得られるような分野では、教育効果の判定 (採点) が可能だと考えられる。すなわち、紙で作成された解答をファイル形式に変換できるか、解答が直接ファイル形式で入力できれば、今回の方法が適用できる。

今回の Word VBA は、筆者の採点作業の軽減の為に、今回の課題専用に作成されている。従って、まだ適用範囲が限られた不完全な VBA なので、より適用範囲の広い VBA の開発 (Word 文章の形式のみならず内容の検証ができる VBA) が必要と考えられる。

参考文献

- 国本温子 (2008). できる大辞典 Excel VBA 2007/2003/2002 対応 インプレスジャパン
- 立山秀利 (2012). 入門者の Excel VBA - 初めての人にベストな学び方 (ブルーバックス) 講談社
- 田中亮 (2013). Excel VBA 逆引き辞典パーフェクト 2013/2010/2007/2003 対応 翔泳社